CLAIMS

1- A method for source decoding a variable-length soft-input codewords sequence (y [1: T]) into a soft-output bit sequence (Λ_v [1: T]), the variable-length soft-input input codewords sequence (y [1: T]) encoded in accordance with a VLC codewords table,

characterized in that it comprises

A. a first stage (100) of implementing a stack decoding algorithm for a sequential estimation of an hard-output bit sequence of said variable length soft-input codewords sequence, including storage of intermediate data contained in the stack and generated by the stack decoding algorithm ; and

B. a second subsequent stage (102) of post-processing the stored intermediate data for generating the soft-output bit sequence (Λ_v [1: T]), a soft-output (Λ(x [t])) being provided for each bit.

2- Method according to claim 1, characterized in that the first stage (100) of implementing the stack decoding algorithm comprises the steps of:

- creating (111) an unitary tree associated with said VLC codewords table, said unitary tree comprising nodes linked by branches, a path being defined by the branches from the initial node to each node of the tree;

- implementing the following sub-steps from an initial node of the unitary tree by using a stack of paths, the stack having a current top path, each path being associated with a cumulative metric, the implementation being carried out until a set of stop conditions is verified:

- Computing (113) a metric M for each branch succeeding the current node of a current top path;
- If (114) the last node of the current top path corresponds to a codeword, concatenate (155) the unitary tree with the current tree by placing the initial node of the unitary tree at least at the last node of the current top path;
- Deleting (116) the current top path from the stack;
- Inserting (117), in the stack, new extended paths made of the current top path and the succeeding branches, the cumulative metric of the new extended path being computed for each extended paths as the metric of current top path increased by the metric of the associated succeeding branch; and

- Selecting (118) a new current top path according to the cumulative metrics associated to the paths.

3- Method according to claim 2, characterized in that the metric associated to a branch leading to a node l at time t is defined as follows:

$$m(l, y[t]) = - \log P(y[t] \mid v(l)) - \log p_t(l) + \log P_0(y[t])$$

- where N_p : the set of nodes having a predecessor ;
- $p_t(l)$ ($l \in N_p$): the a priori probability of the branch reaching the node l at time t;
- $v(l)$ ($l \in N_p$): the value of the branch reaching the node l, $v(l) \in \{0,1\}$;
- $P_0(y[t])$: a Fano - Massey metric which allows to compare fairly sequences of different lengths.

4- A method according to claim 2 or 3, characterized in that the new current top path selected is the path having the smallest cumulative metric among the paths inserted in the stack.

5- A method according to any one of claims 2-4, characterized in that said set of stop conditions comprises the fact that the current top path contains the number of bits and the number of codewords of the variable-length soft-input codewords sequence (y [1: T]).

6- A method according to any one of the preceding claims 2-5, characterized in that the second subsequent stage (102) of post-processing the stored intermediate data comprises the step of approximating each soft-output Λ(x [t]) for each bit by:

$$\Lambda(x[t]) = \mu(t, 0) - \mu(t, 1)$$

where μ(t, 1) is the minimum cumulative metric for all the paths in the stack for which the t estimated bit is 1 and μ(t, 0) is the minimum cumulative metric for all the paths in the stack for which the t estimated bit is 0.

7- A method according to any one of the preceding claims 2-5, characterized in that the second subsequent stage (102) of post-processing the stored intermediate data comprises the step of approximating each soft-output Λ(x [t]) for each bit by:

$$\Lambda(x[t]) = \log \left(\sum_{\substack{1 \le i \le r \\ T_{Pi} >= t \\ \hat{x}_{Pi}[t]=1}} e^{-\mu P_i} \Big/ \sum_{\substack{1 \le i \le r \\ T_{Pi} >= t \\ \hat{x}_{Pi}[t]=0}} e^{-\mu P_i} \right)$$

Where Pi ($i \in [1,\ldots r]$) are the r examined paths stored in the stack and μPi is the cumulative metric of path Pi, T_{Pi} is the length of path Pi and $\hat{x}_{pi}(t)$ is the t^{th} hard bit of an hard bit sequence corresponding to path Pi.

8- A computer program product for a decoder, comprising a set of instructions, which, when loaded into said decoder, causes the decoder to carry out the method claimed in any one of claims 1-7.

9- A computer product for a computer, comprising a set of instructions, which, when loaded into said computer, causes the computer to carry out the method claimed in any one of claim 1-7.

10- A decoder for source decoding a variable-length soft-input codewords sequence (y [1: T]) into a soft-output bit sequence (Λ_v [1: T]),

the variable-length soft-input input codewords sequence (y [1: T]) encoded in accordance with a VLC codewords table,

characterized in that it comprises

means for implementing a stack decoding algorithm for a sequential estimation of an hard-output bit sequence of said variable length soft-input input codewords sequence, including storage means for storing intermediate data contained in the stack and generated by the stack decoding algorithm ; and

means for post-processing the stored intermediate data for generating the soft-output bit sequence (Λ_v [1: T]), a soft-output (Λ(x [t])) being provided for each bit.

11- A Method for iterative decoding a variable-length soft-input sequence (z [1: Tx n/k]) of Tx n/k bits comprising the following steps repeated for each iteration r:

- computing a variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1 : T]) by applying a channel decoding method using as input the variable-length soft-input sequence (z [1: Tx n/k]) and a priori probabilities ratio (ϕ (r - 1) [1: T]) calculated for the previous iteration (r – 1);

- computing a variable-length soft-input codewords sequence ($y^{(r)}$ [t]) depending on the variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1: T]) provided by applying the channel decoding method;

- computing a variable-length soft-output bit sequence ($\Lambda_V^{(r)}$ [t]) by applying a method for source decoding according to any one of claims 1-7 using as input the variable-length soft-input codewords sequence ($y^{(r)}$[t]) depending on the variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1: T]) provided by applying the channel decoding method ; and

- computing the a priori probabilities ratio (ϕ (r) [1: T]) depending on the variable-length soft-input bit sequence ($\Lambda_V^{(r)}$ [t]) applied by said method for source decoding.

12- A receiver for iterative decoding a variable-length soft-input sequence (z [1: Tx n/k]) comprising means for implemented the following steps repeated for each iteration r:

- computing a variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1 : T]) by applying a channel decoding method using as input the variable-length soft-input sequence (z [1: Tx n/k]) and a priori probabilities ratio (ϕ (r - 1) [1: T]) calculated for the previous iteration (r – 1);

- computing a variable-length soft-input codewords sequence ($y^{(r)}$ [t]) depending on the variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1: T]) provided by applying the channel decoding method;

- computing a variable-length soft-output bit sequence ($\Lambda_V^{(r)}$ [t]) by applying a method for source decoding according to any one of claims 1-7 using as input the variable-length soft-input codewords sequence $y^{(r)}[t]$ depending on the variable-length soft-output bit sequence ($\widetilde{\Lambda}_C^{(r)}$ [1: T]) provided by applying the channel decoding method ; a

- computing the a priori probabilities ratio (ϕ (r) [1: T]) depending on the variable-length soft-input bit sequence ($\Lambda_V^{(r)}$ [t]) applied by said method for source decoding.